Preliminary Test Plan

1 Build 1

Build 1 will be the first code improvement. The expected changes will be to parallelize a portion of the code to generate better performance. Build 1 will be on the SGI Origin 3000.

Tests:

1.1 Science Test

- Perform a global, 1/4 deg., 900 sec. dt run using CLM.
- Perform a global, 1/4 deg., 1800 sec. dt run using CLM.
- Perform a global, 1/4 deg., 900 sec. dt run using NOAH.
- Perform a global, 1/4 deg., 1800 sec. dt run using NOAH.
- Perform a global, 5km, 900 sec. dt run using CLM.
- Perform a global, 5km, 1800 sec. dt run using CLM.
- Perform a global, 5km, 900 sec. dt run using NOAH.
- Perform a global, 5km, 1800 sec. dt run using NOAH.

1.2 Restart Test

- Perform a global, 1/4 deg. run writing a "restart file" at the end.
- Perform a continuation run using the above restart file.

1.3 Domain Definition Test

- Identify "special cases" w.r.t. sub-setting code.
- Chose a nominal case.
- For each special case and the nominal case, perform a regional, 1/4 deg. run, using CLM, outputing gridded/interpolated input data.
- For each special case and the nominal case, perform a regional, 5km run, using CLM, outputing gridded/interpolated input data.
- For each special case and the nominal case, perform a regional, 1/4 deg. run, using NOAH, outputing gridded/interpolated input data.
- For each special case and the nominal case, perform a regional, 5km run, using NOAH, outputing gridded/interpolated input data.

1.4 Throughput test

- Perform a global, 5km throughput run using CLM.
- Perform a global, 5km throughput run using NOAH.

Requirements:

Test 1.1, will demonstrate

- LIS General Requirements 3.2, 3.3, 3.4,
- LIS Science Requirements 4.2, 4.4.6, 4.4.11, 4.4.4, 4.4.8,
- LIS Platforms Requirements 7.2, 7.3.

Test 1.2 will demonstrate

- LIS Science Requirment 4.4.7.

Tests 1.1, 1.3 will demonstrate

- LIS Science Requirments 4.1, 4.4, 4.4.12.

Test 1.4 will verify

- LIS Performance Requirment 5.1.

2 Build 2

Build 2 will consist of generating a scheme for interoperability and community requirements and demonstrate a prototype.

Tests:

2.1 Science Test

- Perform a global, 1/4 deg., 900 sec. dt run using CLM on SGI Origin 3000.
- Perform a global, 1/4 deg., 1800 sec. dt run using CLM on SGI Origin 3000.
- Perform a global, 1/4 deg., 900 sec. dt run using NOAH on SGI Origin 3000.
- Perform a global, 1/4 deg., 1800 sec. dt run using NOAH on SGI Origin 3000.
- Perform a global, 1/4 deg., 900 sec. dt run using VIC on SGI Origin 3000.
- Perform a global, 1/4 deg., 1800 sec. dt run using VIC on SGI Origin 3000.
- Perform a global, 5km, 900 sec. dt run using CLM on SGI Origin 3000.
- Perform a global, 5km, 1800 sec. dt run using CLM on SGI Origin 3000.

- Perform a global, 5km, 900 sec. dt run using NOAH on SGI Origin 3000.
- Perform a global, 5km, 1800 sec. dt run using NOAH on SGI Origin 3000.
- Perform a global, 5km, 900 sec. dt run using VIC on SGI Origin 3000.
- Perform a global, 5km, 1800 sec. dt run using VIC on SGI Origin 3000.
- Perform a global, 1/4 deg., 900 sec. dt run using CLM on LIS Linux cluster.
- Perform a global, 1/4 deg., 1800 sec. dt run using CLM on LIS Linux cluster.
- Perform a global, 1/4 deg., 900 sec. dt run using NOAH on LIS Linux cluster.
- Perform a global, 1/4 deg., 1800 sec. dt run using NOAH on LIS Linux cluster.
- Perform a global, 1/4 deg., 900 sec. dt run using VIC on LIS Linux cluster.
- Perform a global, 1/4 deg., 1800 sec. dt run using VIC on LIS Linux cluster.
- Perform a global, 5km, 900 sec. dt run using CLM on LIS Linux cluster.
- Perform a global, 5km, 1800 sec. dt run using CLM on LIS Linux cluster.
- Perform a global, 5km, 900 sec. dt run using NOAH on LIS Linux cluster.
- Perform a global, 5km, 1800 sec. dt run using NOAH on LIS Linux cluster.
- Perform a global, 5km, 900 sec. dt run using VIC on LIS Linux cluster.
- Perform a global, 5km, 1800 sec. dt run using VIC on LIS Linux cluster.

Requirements:

Test 2.1 will verify

- LIS General Requirements 3.2, 3.3, 3.4, 3.5, 3.6, 3.7,
- LIS Science Requirement 4.4.9
- LIS Platforms Requirements 7.5. 7.4.

3 Build 3

Build 3 will make the second code improvements, which will add the database/ data management functions and the visualization functions. This will be the implementation of the GrADS-DODS servers with the internet based user interface. This build will run on the LIS Linux cluster.

Tests:

3.1 Science Test

- Perform a 1/4 deg. run using GEOS, NRL, and AGRMET forcing data; submit run via user interface as a developer.
- Perform a 5km run using GEOS, NRL, and AGRMET forcing data; submit run via user interface as a registered reseacher.

3.2 Restart Test

- Perform a global, 1/4 deg. run writing a "restart file" at the end.
- Perform a continuation run using the above restart file.
- Perform a continuation run using the above restart file over a 5km grid.

3.3 Input Data Retrieval Test

- Perform a run retrieving
  - GEOS data from GrADS-DODS server
  - NRL data from ALMA provider
  - AGRMET shortwave data via ftp
  - AGRMET longwave data via http

3.4 Output Data Conversion Test

- Convert output data from test 3.1 from the GRIB format into a binary format.
- Convert output data from test 3.1 from the GRIB format into the HDF format.
- Convert output data from test 3.1 from the GRIB format into the netCDF format.
- Re-project output data from test 3.1 from the Goode Homolosine projection into the lat/lon projection.
- Re-project output data from test 3.1 from the Goode Homolosine projection into the Lambert-Conformal projection.

3.5 User Interface Test

- Prepare output images from test 3.1.
- View output images via user interface.
- Attempt to access raw output data via user interface as a general public user.
- Access output data via user interface as a registered researcher and as a developer via ftp.
- Access output data via user interface as a registered researcher and as a developer via GrADS-DODS.

Requirements:

Test 3.1 will verify

- LIS Data Management Requirements 8.3, 8.3.4, 8.3.5, 8.4.1, 8.4.3
- LIS Usage Requirement 6.2.3

Test 3.2 will verify

- LIS Science Requirement 4.4.7
- LIS Usage Requirements 6.4 6.5

Tests 3.2 and 3.3 will verify

- LIS Data Management Requirement 8.3.1

Test 3.4 will verify

- LIS Data Management Requirement 8.4.2, 8.4.4

Test 3.5 will verify

- LIS Usage Requirements 6.2.1, 6.2.1.1, 6.2.1.2, 6.2.2.

4 Build 4

Build 4 implement the ESMF compliant version of CLM.

Tests:

Requirements:

5 Build 5

Build 5 is the final version of LIS. LIS will need to run the ESMF compliant version of CLM and the existing LIS/local versions of CLM, NOAH, and VIC.

Tests:

Requirements: